

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:

John F. Poole, et al

Serial No.: 10/709,347

Filed: April 29, 2004

For: Database System with Methodology
for Providing Stored Procedures as Web
Services

Examiner: Seye, Abdou K.

Art Unit: 2194

APPEAL BRIEF

Mail Stop Appeal
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

BRIEF ON BEHALF OF JOHN F. POOLE, ET AL

This is an appeal from the Final Rejection mailed August 28, 2007, in which currently-pending claims 1-40 stand finally rejected. Appellant filed a Notice of Appeal on December 3, 2007. This brief is submitted electronically in support of Appellant's appeal.

TABLE OF CONTENTS

1.	REAL PARTY IN INTEREST	3
2.	RELATED APPEALS AND INTERFERENCES	3
3.	STATUS OF CLAIMS.....	3
4.	STATUS OF AMENDMENTS.....	3
5.	SUMMARY OF CLAIMED SUBJECT MATTER.....	4
6.	GROUND OF REJECTION TO BE REVIEWED.....	6
7.	ARGUMENT	7
	A. First Ground: Claims 1-11, 13-31 and 33-40 rejected under 35 U.S.C. 102(e)	7
	B. Second Ground: Claims 12 and 32 rejected under 35 U.S.C. 103(a).....	12
8.	CLAIMS APPENDIX	16
9.	EVIDENCE APPENDIX	21
10.	RELATED PROCEEDINGS APPENDIX.....	22

1. REAL PARTY IN INTEREST

The real party in interest is assignee Sybase, Inc. located at One Sybase Drive, Dublin, CA 94568.

2. RELATED APPEALS AND INTERFERENCES

There are no appeals or interferences known to Appellant, the Appellant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

3. STATUS OF CLAIMS

The status of all claims in the proceeding is as follows:

Rejected: Claims 1-40

Allowed or Confirmed: None

Withdrawn: None

Objected to: None

Canceled: None

Identification of claims that are being appealed: Claims 1-40

An appendix setting forth the claims involved in the appeal is included as Section 8 of this brief.

4. STATUS OF AMENDMENTS

One Amendment has been filed in this case. Appellant filed an Amendment on June 18, 2007 in response to a non-final Office Action dated February 23, 2007. In the Amendment filed on June 18, 2007, the pending claims were amended in a manner which Appellant believes clearly distinguished the claimed invention over the art of record, for overcoming the art rejections. In response to the Examiner's Final Rejection dated August 28, 2007 (hereinafter "Final Rejection") finally rejecting Appellant's claims, Appellant filed a Notice of Appeal. Appellant has chosen to forego filing an Amendment After Final as it is believed that further amendments to the claims are not warranted in

view of the art. Accordingly, no amendments have been entered in this case after the date of the Final Rejection.

5. SUMMARY OF CLAIMED SUBJECT MATTER

As to Appellant's **First Ground** for appeal, Appellant asserts that the art rejection of Appellant's **claims 1-11, 13-31 and 33-40** under 35 USC Section 102(e) relying on U.S. Published Application 20040199636 of Brown et al ("Brown") fails to teach or suggest all of the claim limitations of Appellant's rejected claims 1-11, 13-31 and 32-40, where the claimed invention is set forth in the embodiment in **independent claim 1**: In a database system (Appellant's specification paragraphs [0012]-[0013], paragraph [0063]; Fig. 4 at 410; Fig. 3 at 340), a method for providing a stored procedure as a Web service (Appellant's specification paragraphs [0012]-[0013], paragraph [0063], paragraphs [0121]-[0123]; Figs. 5A-B), the method comprising: predefining a stored procedure to be invoked upon receiving a client request for a particular Web service (Appellant's specification paragraph [0012], paragraph [0062] (developer authors stored procedure), paragraph [0061] (mapping table that maps given client request to particular stored procedure), paragraph [0075]), receiving an incoming request from a particular client for the particular Web service at an HTTP server incorporated into the database system (Appellant's specification paragraph [0012], paragraphs [0061]-[0062], paragraph [0065] (Requests coming in handed off to HTTP server which resides in the same executable space as the database engine), paragraphs [0121]-[0122]; Fig. 5A at 501-503; Fig. 4 at 431), in response to the incoming request, identifying the stored procedure that is predefined for the particular Web service (Appellant's specification paragraph [0012], paragraph [0061] (mapping table that maps given client request to particular stored procedure), paragraph [0122], paragraphs [0139]-[0140]; Fig. 5B at 508; Fig. 6 at 607; also, see generally paragraphs [0096]-[0106]) executing the identified stored procedure for generating a result set (Appellant's specification paragraph [0012], paragraph [0061], paragraphs [0065]-[0067], paragraph [0123]; Fig. 5B at 509-510; Fig. 6B at 610-611), and returning the result set back to the particular client (Appellant's specification paragraph [0012], paragraph [0061], paragraphs [0067]-[0068], paragraph [0137]; Fig.

5B at 510-515).

For Appellant's argument under the **First Ground** for appeal, Appellant additionally argues that the art rejection under 35 USC Section 102(e) relying on Brown (above) fails to teach or suggest all of the claim limitations of Appellant's rejected claims, where the claimed invention is set forth in the embodiment in **independent claim 21**: A database system (Appellant's specification paragraphs [0012]-[0013], paragraph [0063]; Fig. 4 at 410; see generally Fig. 3) providing stored procedures as Web services (Appellant's specification paragraphs [0012]-[0013], paragraphs [0121]-[0123]; Figs. 5A-B), the system comprising: a database engine stored on a computer-readable medium, which when executed controls a database (Appellant's specification paragraph [0013], paragraph [0063], paragraph [0066]; Fig. 3 at 360; Fig. 4 at 450; also see generally, paragraphs [0046]-[0059]) that includes a stored procedure to be invoked upon receiving a request for a particular Web service (Appellant's specification paragraph [0013], paragraph [0062] (developer authors stored procedure), paragraph [0061] (mapping table that maps given client request to particular stored procedure), paragraph [0075]), a communications layer of the database system for receiving an incoming request from a particular client for the particular Web service (Appellant's specification paragraph [0013], paragraph [0063], paragraph [0064] (communications layer handles incoming requests at the database server), paragraph [0121]; Fig 4 at 420; Fig. 5A at 501-502), an HTTP server incorporated into the database system for parsing and validating the incoming request (Appellant's specification paragraph [0013], paragraphs [0061]-[0062], paragraph [0065] (Requests coming in handed off to HTTP server which resides in the same executable space as the database engine), paragraph [0122]; Fig. 4 at 431; Fig. 5A at 503-504; Fig. 6 at 602-604), a request layer of the database system for identifying the stored procedure corresponding to the particular Web service and executing the identified stored procedure for the particular Web service, for generating a result set (Appellant's specification paragraph [0012], paragraph [0061] (mapping table that maps given client request to particular stored procedure), paragraphs [0065]-[0067], paragraph [0123] (request layer executes target stored procedure specified for requested Web service), paragraphs [0139]-[0140]; Fig. 4 at 440; Fig. 5B at 508-510; Fig. 6 at 610), and a presentation layer of the database system for returning the result set back to the particular

client (Appellant's specification paragraph [0013], paragraph [0061], paragraphs [0067]-[0068], paragraph [0137]; Fig. 4 at 433; Fig. 5B at 510-515).

For Appellant's argument under the **First Ground** for appeal, Appellant additionally argues based on **dependent claim 2** which includes limitations that the HTTP server is built into the database engine of the database system (Appellant's specification, paragraph [0061], paragraph [0065] (the HTTP server resides in the same executable space as the database engine); Fig. 4 at 431). For Appellant's argument under the **First Ground** for appeal, Appellant further argues based on **dependent claim 22** which includes limitations that the HTTP server is built into the same executable space as the database engine (Appellant's specification, paragraph [0061], paragraph [0065] (the HTTP server resides in the same executable space as the database engine); Fig. 4 at 431).

As to Appellant's **Second Ground** for appeal, Appellant asserts that the art rejection under **Section 103(a)** relying on the combination of Brown (above) and U.S. Published Application 20030191769 of Crisnan et al ("Crisnan") fails to teach or suggest all of the claim limitations of Appellant's **rejected claims 12 and 32**, where the claimed invention is set forth in the embodiment in **independent claims 1 and 21** (the mapping of which is shown above under Appellant's **First Ground** for appeal, and which hereby is incorporated by reference). For Appellant's argument under the **Second Ground** for appeal, Appellant additionally argues based on **dependent claims 12 and 32** which include limitations the execution of the database stored procedure may occur asynchronously with respect to the incoming client requests for the Web service (Appellant's specification paragraph [0123], paragraph [0139] (requests queued to be processed), paragraph [140] (queued request picked up), paragraph [0181]; Fig. 6A at 605-606).

6. GROUNDS OF REJECTION TO BE REVIEWED

The grounds for appeal are:

(1st) Whether claims 1-11, 13-31 and 33-40 are unpatentable under 35 U.S.C. Section 102(e) as being anticipated by U.S. Published Application 20040199636 of Brown et al (hereinafter "Brown"); and

(2nd) Whether claims 12 and 32 are unpatentable under 35 U.S.C. Section 103(a) as obvious over Brown (above) in view of U.S. Published Application 20030191769 of Crisnan et al (hereinafter "Crisnan").

7. ARGUMENT

A. First Ground: Claims 1-11, 13-31 and 33-40 rejected under 35 U.S.C. 102(e)

1. General

Under Section 102, a claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in the prior art reference. As discussed in detail below, the Brown reference fails to teach each and every element set forth in independent claims 1 and 21, as well as other dependent claims thereof and therefore fails to establish anticipation of the claimed invention under Section 102.

2. Claims 1, 2-11 and 13-21, 23-31 and 33-40

The Examiner has rejected 1-11, 13-31 and 33-40 as being anticipated by Brown. The following rejection of Appellant's claim 1 and 19-21 by the Examiner is representative of the Examiner's rejection of the Appellant's claims as being anticipated by Brown:

Claims 1, 19-21, Brown teaches, in a database system, a method system and product for providing a stored procedure as a Web service, the method comprising:
predefining a stored procedure to be invoked upon receiving a client request for a particular Web service (paragraph 28);
receiving an incoming request from a particular client for the particular Web Service at an HTTP server incorporated into the database system (abstract; Fig. 2; paragraph 26-30; the database system is the service provider 11 of Fig. 2);
in response to the incoming request, identifying the stored procedure that is predefined for the particular Web service (paragraph 28, 45 and 58);
executing the identified stored procedure for generating a result set; and returning the result set back to the particular client (Fig. 6; paragraph

(Final Rejection, paragraph 3, pages 2-3)

At the outset, Appellant does not claim to have invented the notion of a three-tier

architecture in which a Web server (or application server) receives requests from client devices and interacts with a back-end database server to obtain and serve up information to the clients. Instead, Appellant's invention removes the need for a middle-tier Web server (or other program logic external to the database server) by including a lightweight HTTP server and supporting infrastructure for responding to client HTTP requests directly into the database server itself (Appellant's specification paragraphs [0061] - [0063]; Fig. 4 at 431). In fact, with Appellant's solution, the HTTP server resides in the same executable space as the database engine (Appellant's paragraph [0065]).

With Appellant's invention Web clients communicate directly with the database server without the need for an intermediary Web server or other external program logic. As shown at Appellant's Fig. 4, client(s) 401 communicate directly with the database server 410, which includes an HTTP server 431 (Appellant's specification, paragraph [0063]). When client requests are received at the database server, the built-in HTTP server translates incoming client requests into stored procedure calls, complete with database features such as authentication (Appellant's specification, paragraph [0061]). The client requests are mapped to corresponding database stored procedures which have been previously defined and implemented. The solution converts client requests from HTTP into a format which can be used by the database engine and the stored procedure is executed (Appellant's specification, paragraph [0066]). The results from execution of the stored procedure are then formatted for return to the client (Appellant's specification, paragraph [0067]). For example, the results may be formatted as a SOAP response or an XML document (Appellant's specification, paragraph [0067]). The results are then returned to the client over the HTTP connection in the appropriate format.

These features are specifically included in Appellant's claims. For example, Appellant's claim 1 includes the following claim limitations:

In a database system, a method for providing a stored procedure as a Web service, the method comprising:
predefining a stored procedure to be invoked upon receiving a client request for a particular Web service;
receiving an incoming request from a particular client for the particular Web service at an HTTP server incorporated into the database system;
in response to the incoming request, identifying the stored procedure that is predefined for the particular Web service;

executing the identified stored procedure for generating a result set; and returning the result set back to the particular client.

(Appellant's claim 1, emphasis added)

The Examiner references Brown at Fig. 2, paragraphs 26-30 and abstract as including comparable teachings of an HTTP server incorporated into the database system; however review of the teachings of Brown referenced by the Examiner (as well as the balance of the reference) finds that Brown's solution does not, in fact, incorporate an HTTP server into the database server as discussed below.

Fig. 2 of Brown depicts a Service Provider 11 which includes three components which are depicted as separate entities from each other: an HTTP Server 25, SOAP Server 27 and Database 29. As shown at Fig. 2, the Database 29 is separate from the HTTP Server 25. The HTTP Server 25 is not incorporated as part of Brown's Database server (Database 29) (Brown, Fig. 2). This is confirmed by the fact that that the HTTP Server 25 communicates with the Database 29 through another component; namely, the SOAP Server 27 (Brown, Fig. 2). As shown at Fig. 2 of Brown, the HTTP Server 25 receives an XML message 31 from an external source (Service requestor 15) and interacts with the SOAP Server 27. The SOAP Server 27, in turn interacts with the Database 29. Thus, Brown's HTTP and Database Servers are separate components with a third entity - the SOAP server – interposed as an intermediary between the two.

The fact that the database server and HTTP server are separate components is also evident from the referenced text included at paragraphs [0026]-[0030] of Brown. Paragraph [0028] of Brown, for example, provides as follows:

DXX is a collection of stored procedures, user defined types (UDTs), and user defined functions (UDFs) that are typically used in programs written in SQL, Java, C++, and other languages. The DXX invoker handles universal resource locators (URLs) having the extensions ".dadx" that reference web services and ".dtd" that reference document type definition (DTD) documents. Specifically, the DXX invoker provides runtime support for invoking DADx documents as web services written, e.g., in Apache Simple Object Access Protocol (SOAP). The servlet of the router connects to a database at invocation times. Preferably, this connection is established by a Java database connection (JDBC).

(Brown, paragraph [0028], emphasis added)

As illustrated above, Brown discusses using HTTP servers (e.g., Apache) external to the Database Server as well as connecting to the Database server (e.g., preferably the connection is established by JDBC). Paragraph [0029] of Brown also confirms that this is the case as it provides as follows:

In one possible embodiment, the DXX Invoker, a Java component, interfaces with Apache SOAP 2.1 runtime using a pluggable provider support. A web application developer creates an instance of the DXX Invoker for each database that is to be accessed. Each instance of the DXX Invoker is associated with a database connection and a set of DADx files. The DXX Admin servlet is provided to simplify the task of creating instances of the DXX Invoker. Specifically, the DXX Admin servlet provides a web user interface for configuring instances of the DXX Invoker. Someone, e.g., a database administrator, sets up the databases and enables them for use by DXX. The service provider creates DAD and DADx documents and deploys them to the web application. Each DADx document is associated with a URL that identifies a specific web service.

(Brown paragraph [0029], emphasis added)

As illustrated above, with Brown's solution a Web application may connect to multiple databases and the Web application developer creates an instance of the "DXX Invoker" for each such database. Furthermore, the developer associates a database connection (and set of DADx files) with each such instance of the DXX Invoker, thereby making it very clear that the HTTP server is not integrated into the database server.

Fig. 6 of Brown provides additional evidence that the HTTP server of Brown's solution is not incorporated into the database system. As shown, clients connect to a Web Application Server 63 which is separate from the Database 29, with DADX 51 and XML Extender DAD 37 interposed between the Database 29 and the Web Application Server 63 (Brown, Fig. 6). This is in direct contrast with Appellant's Fig. 4 in which the clients 401 interact directly with the database system 410 and which also clearly shows the HTTP server included as a component of the database system (see also, Appellant's specification, paragraph [0063]; Fig. 4 at 401, 410, 431). These figures clearly illustrate the differences between Appellant's intervention and that of Brown. Brown simply describes a conventional three-tier solution in which an application server (e.g., Web Application Server 63 as well as other components such as the DADX 51 and XML Extender DAD 37 as shown at Fig. 6 of Brown) is interposed between clients (e.g., Soap

client 61a) and the database (e.g., Database 29). Appellant's invention, in contrast, provides for clients to directly interact with the database by providing an integrated solution which includes support for Web services inside the database system itself. With Appellant's solution, users are not required to set up a variety of disparate components external to the database, but rather support for processing of Web services is integrated into the database system.

The final portion of Brown referenced by the Examiner as providing teachings of incorporating the HTTP Server into the database system is the Abstract which states as follows:

The present invention provides a method, system and program product for automatically generating an invoking mechanism from a web service description, thereby enabling integration of a **service external to a database** into a database such that the service may be easily invoked **from the database**. Preferably, the service is a web service available over the internet. The service may be invoked from any of a number of invoking mechanisms of the database. In a first specific embodiment, the mechanism comprises a user-defined function within an SQL statement. In a second specific embodiment, the mechanism comprises a virtual table. In a third specific embodiment, the mechanism comprises a stored procedure. In a fourth specific embodiment, the mechanism comprises a trigger. In a fifth specific embodiment, the mechanism comprises a federated table accessed via a nickname and implemented using a wrapper.

(Brown Abstract, emphasis added)

As illustrated above, here Brown in fact appears to talking about something fundamentally different than Appellant's claimed invention; namely, the database providing a mechanism for invoking a Web service that is external to the database. In other words, in this aspect of Brown's invention the database is essentially acting as the client in invoking an external Web service. Appellant respectfully believes that this feature is not at all analogous to Appellant's claimed invention.

3. Claims 2 and 22

Appellant's claim 2 includes further limitations that the HTTP server is built into (resides in the same executable space as) the database engine of the database system (Applicant's specification, paragraph [0065]). Similarly, claim 22 provides that the HTTP server is built into the same executable space as the database engine (Applicant's

specification, paragraph [0065]). The Examiner again references Brown at Abstract and Fig 2:25 for the corresponding teachings. However, as described above in detail Brown's solution provides for the database server and HTTP server to be separate components with an intermediary between them. Therefore as Brown's HTTP server is a separate component from the database server, it is not incorporated into the database engine as provided in Appellant's claim 22, nor does it reside in the same executable space as the database engine as provided in Appellant's claim 22.

4. Conclusion

As described in detail above, Brown describes a conventional three-tier solution in which an application server is interposed between clients and a database. Appellant's solution, in contrast, incorporates an HTTP server inside the database system itself, so that the database system itself includes support for Web services without having to rely on external components. Therefore, as Brown does not teach or suggest all of the claim limitations of Appellant's claims 1-11, 13-31 and 33-40 it is respectfully submitted that the claims distinguish over this reference and the Examiner's rejection of these claims under Section 102 should not be sustained.

B. Second Ground: Claims 12 and 32 rejected under 35 U.S.C. 103(a)

1. General

Under Section 103(a), a patent may not be obtained if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. To establish a prima facie case of obviousness under this section, the Examiner must establish: (1) that there is some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings, (2) that there is a reasonable expectation of success, and (3) that the prior art reference (or references when combined) must teach or suggest all the claim limitations. (See e.g., MPEP 2142). The reference(s) cited by the Examiner fail to meet these conditions.

2. Claims 12 and 32

Claims 12 and 32 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Brown (above) in view of Crisnan. As to these claims, the Examiner continues to rely on Brown as substantially teaching Appellant's invention, but acknowledges that Brown does not teach “wherein execution of the identified stored procedure occurs asynchronously with respect to the incoming request” (Final Rejection, paragraph 5, page 7). The Examiner therefore adds Crisnan for these teachings.

Claims 12 and 32 are dependent upon Appellant's independent claims 1 and 21 and, therefore, are believed to be allowable for at least the reasons cited above under Appellant's **First Ground** of appeal (incorporated by reference herein) pertaining to the deficiencies of Brown in respect to Appellant's invention. As Crisnan does not provide any teaching of incorporating an HTTP server into a database system, it does not cure the deficiencies of Brown as to Appellant's invention. Claims 12 and 32 are also believed to be patentable for the following additional reasons.

As described above, Appellant's claimed invention provides for representing Web services as database stored procedures and including support for receiving and responding to client requests (i.e., HTTP requests) for Web services inside the database system itself. When an incoming request for a Web service is received at the HTTP server incorporated into the database system of Appellant's invention, the Web service being invoked is identified and a cursor is created to execute the target database stored procedure associated with this Web service (Appellant's specification paragraphs [0122]-[0123]). The target stored procedure for the requested Web service is then executed (Appellant's specification paragraph [0123]; Fig. 5B at 509). Appellant's claims 12 and 32 provide that the execution of this database stored procedure may occur asynchronously with respect to the incoming client requests for the Web service as the incoming client requests are queued up at the HTTP server and processed asynchronously (Appellant's specification paragraph [0123], paragraph [0139] (requests queued to be processed), paragraph [140] (queued request picked up), paragraph [0181]; Fig. 6A at 605-606).

Crisnan, in contrast, describes asynchronous execution of a workflow function external to the database which is called by a SQL program or trigger. As shown at Fig.

11 of Crisnan, for example, a SQL Program (or trigger) 202 calls User Defined Functions (UDF) 208 a, b, n external to the database (Crisnan, Fig. 11 and paragraph [0132]). The UDF, in turn, invokes a workflow defined within a separate Workflow Engine 210 (Crisnan, Fig. 11 and paragraph [0132]). As Crisnan's Workflow Engine is external to the database and is, in fact, called by a database program or trigger through a User Defined Function, it is distinguishable from Appellant's claimed invention. Appellant's invention involves a different situation in which incoming client requests (HTTP requests) for Web services are received at a database server. These incoming requests are queued by an HTTP server incorporated into the database server and the associated database stored procedures are executed in the database system asynchronously.

3. Conclusion

As discussed in detail above, neither Brown nor Crisnan include any teaching of incorporating an HTTP server into a database system. Moreover, neither reference includes the specific teaching of a database system executing a database stored procedure asynchronously with respect to an incoming client request for the Web service represented by such stored procedure. Therefore, as the two references, even when combined, do not teach or suggest all the limitations of Appellant's claims it is respectfully submitted that Appellant's claimed invention is distinguishable over the prior art and that the Examiner's rejection under Section 103 should not be sustained.

C. Conclusion

The present invention greatly improves the efficiency of developing and implementing Web services by providing an integrated solution which provides support for processing of Web services inside a database system, thereby enabling the database system to respond to requests for Web services without having to rely on external components (e.g., a Web application server). It is respectfully submitted that the present invention, as set forth in the pending claims, sets forth a patentable advance over the art.

In view of the above, it is respectfully submitted that the Examiner's rejection of Appellant's claims under 35 U.S.C. Section 102 and 35 U.S.C. Section 103 should not be sustained. If needed, Appellant's undersigned attorney can be reached at 925 465 0361. For the fee due for this Appeal Brief, please refer to the attached Fee Transmittal Sheet. This Appeal Brief is submitted electronically in support of Appellant's Appeal.

Respectfully submitted,

Date: January 28, 2008

/G. Mack Riddle/

G. Mack Riddle, Reg. No. 55,572
Attorney of Record

925 465-0361
925 465-8143 FAX

8. CLAIMS APPENDIX

1. In a database system, a method for providing a stored procedure as a Web service, the method comprising:

predefining a stored procedure to be invoked upon receiving a client request for a particular Web service;

receiving an incoming request from a particular client for the particular Web service at an HTTP server incorporated into the database system;

in response to the incoming request, identifying the stored procedure that is predefined for the particular Web service;

executing the identified stored procedure for generating a result set; and
returning the result set back to the particular client.

2. The method of claim 1, wherein the incoming request is received by an HTTP server that is built into a database engine of the database system.

3. The method of claim 1, further comprising:

upon receiving the incoming request, verifying that the request comprises a valid HTTP request for a Web service.

4. The method of claim 1, wherein client requests employ HTTP protocol.

5. The method of claim 4, wherein client requests further specify a selected one of XML, SOAP, WSDL, and raw format.

6. The method of claim 1, wherein said returning step further comprises:
formatting the result set into a particular presentation format; and thereafter
returning the formatted result set back to the particular client.

7. The method of claim 1, wherein said executing step further comprises:
creating a temporary pseudo connection to a database engine of the database

system; and

executing the stored procedure through said temporary pseudo connection.

8. The method of claim 1, wherein the incoming request comprises a URL.

9. The method of claim 8, wherein the URL includes parameter information affecting how the identified stored procedure is executed.

10. The method of claim 1, wherein the identified stored procedure may include any valid SQL statement.

11. The method of claim 1, wherein the identified stored procedure itself may invoke other stored procedures.

12. The method of claim 1, wherein execution of the identified stored procedure occurs asynchronously with respect to the incoming request.

13. The method of claim 1, wherein the identified stored procedure selects data from a database, and wherein the result set returned to the particular client comprises that data formatted in a manner suitable for return via HTTP protocol.

14. The method of claim 13, wherein the data is formatted for return as XML-formatted data.

15. The method of claim 14, wherein the result set comprises a plurality of database rows, and wherein the XML-formatted data comprises said plurality of database rows delimited with XML row tags.

16. The method of claim 1, wherein the identified stored procedure itself may set HTTP header information that is returned to the particular client.

17. The method of claim 1, wherein the incoming request is received via a selected one of HTTP, FTP, and telnet protocol.

18. The method of claim 1, wherein the system first checks user authentication for the particular client before executing the identified stored procedure.

19. A computer-readable medium having processor-executable instructions for performing the method of claim 1.

20. The method of claim 1 further comprising:
A downloadable set of processor-executable instructions stored on a computer-readable medium for performing the method of claim 1.

21. A database system providing stored procedures as Web services, the system comprising:

a database engine stored on a computer-readable medium, which when executed controls a database that includes a stored procedure to be invoked upon receiving a request for a particular Web service;

a communications layer of the database system for receiving an incoming request from a particular client for the particular Web service;

an HTTP server incorporated into the database system for parsing and validating the incoming request;

a request layer of the database system for identifying the stored procedure corresponding to the particular Web service and executing the identified stored procedure for the particular Web service, for generating a result set; and

a presentation layer of the database system for returning the result set back to the particular client.

22. The system of claim 21, wherein the HTTP server resides in an executable space that is shared with the database engine of the database system.

23. The system of claim 22, wherein the HTTP server verifies that the incoming request comprises a valid HTTP request for a Web service.

24. The system of claim 21, wherein client requests employ HTTP protocol to communicate with the database system.

25. The system of claim 24, wherein client requests further specify a selected one of XML, SOAP, WSDL, and raw format.

26. The system of claim 21, wherein the presentation layer formats the result set into a particular presentation format before returning the result set to the particular client.

27. The system of claim 21, wherein the request layer includes program logic for creating a temporary pseudo connection to the database engine of the database system, and for executing the stored procedure through the temporary pseudo connection.

28. The system of claim 21, wherein the incoming request comprises a URL.

29. The system of claim 28, wherein the URL includes parameter information affecting how the identified stored procedure is executed.

30. The system of claim 21, wherein the identified stored procedure may include any valid SQL statement.

31. The system of claim 21, wherein the identified stored procedure itself may invoke other stored procedures.

32. The system of claim 21, wherein execution of the identified stored procedure occurs asynchronously with respect to the incoming request.

33. The system of claim 21, wherein the identified stored procedure selects data

from the database, and wherein the result set returned to the particular client comprises that data formatted in a manner suitable for return via HTTP protocol.

34. The system of claim 33, wherein the data is formatted for return as XML-formatted data.

35. The system of claim 34, wherein the result set comprises a plurality of database rows selected from the database, and wherein the XML-formatted data comprises said plurality of database rows delimited with XML row tags.

36. The system of claim 21, wherein the identified stored procedure itself may set HTTP header information that is returned to the particular client.

37. The system of claim 21, wherein the incoming request is received via a selected one of HTTP, FTP, and telnet protocol.

38. The system of claim 21, wherein the system first checks user authentication for the particular client before executing the identified stored procedure.

39. The system of claim 21, wherein the request layer includes program logic for translating the request into a form that can be processed by the database engine.

40. The system of claim 21, wherein the incoming request is received over an HTTP connection, and the result set is returned to the particular client over an HTTP connection.

9. EVIDENCE APPENDIX

This Appeal Brief is not accompanied by an evidence submission under §§ 1.130, 1.131, or 1.132.

10. RELATED PROCEEDINGS APPENDIX

Pursuant to Appellant's statement under Section 2, this Appeal Brief is not accompanied by any copies of decisions.